

## CLAIMS

Please amend the claims as shown in the following claim listing:

1. (Currently amended) A logic circuit comprising:  
a control flow logic to select and fetch a trace descriptor for processing, the fetched trace descriptor including at least one dependency descriptor, the dependency descriptor including dependency information for an instruction sequence and a location of the instruction sequence;  
and  
a data flow logic coupled to the control flow logic to execute the instruction sequence according to the dependency information ~~stored~~ in the dependency descriptor.
2. (Currently amended) The logic circuit of claim 1 ~~further~~ comprising a ~~first~~ storage area coupled to the control flow logic and the data flow logic, the ~~first~~ storage area to store the dependency descriptor ~~after separation from a selected~~ the fetched trace descriptor by the control flow logic.
3. (Currently amended) The logic circuit of claim ~~2~~ 1 ~~further~~ comprising a ~~second~~ storage area coupled to the control flow logic, the ~~second~~ storage area ~~dedicated to storage of~~ store trace descriptors.

4. (Currently amended) The logic circuit of claim ~~3~~1 comprising a ~~third~~ storage area coupled to the data flow logic, the ~~third~~ storage area to store instructions contiguously based on dependency information.
5. (Currently amended) The logic circuit of ~~claim 4~~claim 1 comprising a ~~fourth~~ storage area coupled to the data flow logic and control flow logic, the ~~fourth~~ storage area to store live-out data.
6. (Currently amended) The logic circuit of claim ~~5~~1 comprising a ~~fifth~~ storage area coupled to the control flow logic, the ~~fifth~~ storage area to map live-in and live-out data.
7. (Canceled).
8. (Currently amended) The logic circuit of claim 1 wherein the trace descriptor includes aggregate live-in data for a plurality of dependency descriptors in the trace ~~description~~ descriptor.
9. (Currently amended) The logic circuit of claim 1 wherein the trace descriptor includes aggregate live-out data for a plurality of dependency ~~descriptor~~ descriptors in the trace ~~description~~ descriptor.

10. (Currently amended) A computer system comprising:
- at least one memory device to store trace descriptors and instruction sequences;
  - a bus coupled to the at least one memory device;
  - a control flow logic device ~~to analyze dependencies among instruction sequences in a trace and create a trace descriptor comprising a dependency descriptor indicating a live-in value for an instruction sequence~~ to select and fetch one of the trace descriptors, the fetched trace descriptor including a plurality of dependency descriptors having location and dependency information for corresponding instruction sequences; and
  - a data flow logic device coupled to the control flow logic device to receive a dependency descriptor dispatched from the control flow logic device, to fetch an instruction sequence corresponding to the received dependency descriptor, and to execute a plurality of the instruction sequences the fetched instruction sequence according to the dependency information stored in the dependency descriptor.
11. (Currently amended) The computer system of claim 10 ~~further comprising a dependency claim~~ an issue window coupled between the control flow logic device and the data flow logic device, the dependency claim issue window to store the dependency descriptor dispatched from the control flow logic device.
12. (Canceled).

13. (Currently amended) The computer system of claim ~~12~~ further comprising a ~~third storage area coupled to the data flow logic device, the third storage area to store instructions~~ 10 wherein the at least one memory unit is to store an instruction sequence contiguously based on dependency information.

14. (Currently amended) The computer system of claim ~~13~~ further 10 comprising a ~~fourth~~ storage area coupled to the data flow logic device and control flow logic device, the ~~fourth~~ storage area to store live-out data.

15. (Currently amended) The computer system of claim ~~14~~ further 10 comprising a ~~fifth~~ storage area coupled to the control flow logic, the ~~fifth~~ storage area to map live-in and live-out data.

16. (Canceled).

17. (Currently amended) The computer system of claim 10 wherein the fetch trace descriptor includes aggregate live-in data for a plurality of dependency descriptors in the fetch trace ~~description~~ descriptor.

18. (Currently amended) The computer system of claim 10 wherein the fetch trace descriptor includes aggregate live-out data for a ~~plurality of~~ dependency descriptors in the fetch trace ~~description~~ descriptor.

19. (Currently amended) The computer system of claim 10 wherein dependency information of the received dependency descriptor includes a ~~live-out value for the instruction sequence~~ live-out data.

20. (Currently amended) A method of processing instructions comprising:  
selecting and fetching a trace descriptor in accordance with program control flow;  
~~separating out~~ identifying from the fetched trace descriptor a dependency descriptor including dependency information for a set of instructions ~~from the trace descriptor~~ and a location of the set of instructions;  
fetching the set of instructions ~~at a location indicated by~~ from the location in the dependency descriptor; and  
executing a ~~plurality of the instruction sequences~~ the set of instructions according to ~~dependencies stored~~ the dependency information in the dependency descriptor.

21. (Currently amended) A method according to claim 20 ~~further~~ comprising:  
updating live-out data in a ~~first~~ storage area.

22. (Currently amended) A method according to claim ~~21~~ 20 comprising:  
storing the identified dependency descriptor ~~extracted by~~ from a control flow logic into a ~~second~~ storage area; and  
reading the dependency descriptor out of the ~~second~~ storage area into a data flow logic.
23. (Currently amended) A method according to claim ~~22~~ 20 wherein the fetching of ~~a~~ the set of instructions is completed just in time for execution.
24. (Currently amended) A method according to claim ~~23~~ 20 wherein the executing comprises executing instructions ~~are~~ out of order.
25. (Currently amended) A method according to claim ~~24~~ 21 comprising:  
updating the architectural state using ~~the~~ data in the ~~first~~ storage area.
26. (Currently amended) A method according to claim ~~25~~ 21 comprising:  
recovering an earlier architectural state after a misprediction using ~~the~~ data in the ~~first~~ storage area.
27. (Currently amended) A method according to claim ~~20~~ 21 wherein the selecting ~~involves~~ comprises predicting ~~the~~ a next trace descriptor to process.

28. (Currently amended) A machine-readable medium that provides instructions, which when executed by a machine cause the machine to perform operations comprising:

selecting and fetching a trace descriptor in accordance with program control flow;  
~~separating out~~ identifying from the fetched trace descriptor a dependency descriptor including dependency information for a set of instructions ~~from the trace descriptor~~ and a location of the set of instructions;

storing the dependency descriptor in an issue window to await assignment to an execution unit;

fetching the set of instructions ~~described by~~ from the location in the dependency descriptor; and

executing the set of ~~the~~ instructions according to the ~~dependencies stored~~ dependency information in the dependency descriptor.

29. (Currently amended) The machine-readable medium of claim 28, wherein the operations ~~further~~ comprise:

updating live-out data in a ~~first~~ storage area.

30. (Currently amended) The machine-readable medium of claim ~~29~~ 28, wherein the ~~operation further~~ operations comprise:

reading the dependency descriptor out of the issue window into ~~the~~ data flow logic.